News categorization

Vikash Kumar Pandey
MT18086

*A. **Keywords***

Text categorization, classification, support vector machines, case-folding, lemmetization.

I. Introduction

The growth of information on the internet and corporate worlds is continuously increasing, and the users want to access this data so it can easily find, filter and manage the resources. The aim of news categorisation is the Text classification of news document to the class,to which it belongs. An article can be in multiple categories or none of the types. Using classifier the goal is to learn the classifiers from examples so that it can assign the class automatically. That's, why it is being called supervised learning algorithm as a model, is trained in supervision with the help of examples. The first step in news categorisation is to collect the data set. The data-set is collected from various resources available on the internet, and we also crawl the data from multiple websites using the beautiful soup library[2] (used for extract the data from HTML file). The data is now to be converted into a suitable format suited for each of the classifications algorithms which can be typically a list of string. But the problem is that the data which is crawled is in the form of garbage (unordered data+punctuation), so we have to preprocess our data set to do this we apply lemmatization, punctuation removal techniques, stop-words to remove useless (is, are, the etc..) words, case-folding and some other preprocessing method to clean our data-set.

After preprocessing we extract features by Term frequency and Inverse document frequency (TF-IDF) and by count vectorizer and compare both with different classifiers, and we got features selection techniques TF-IDF to perform well because the words which are common in most of the documents are supposed to be not a useful feature. IDF computes the logarithmically scaled inverse fraction of the total number of documents by the total number of documents containing that word IDF(t) = log(total number of documents/total number of documents containing t). The method is proposed in [1] and we used it for features selection. After getting feature vector which is going to be the input of our classifier and this feature vector which is computed by the TF-IDF, perform very well because in count vectorizer each message is separated into tokens and the number of times each token occurs in a message is counted. But in TF-IDF, TF count how often a word occurs in a document if we have several occurrences of the same word in one document we can expect the TF-IDF get increased. IDF is representing how common a word is across documents,if a word used in many documents then the TF-IDF will decrease which is a good sign for features selection but count vectorizer is not handling that's why when we apply TF-IDF with SVM , and it correctly classifies the category corresponding to given data with the accuracy of 93.7%. We apply various classification techniques like Naive Bayes, Linear discriminant analysis, and Random forest with both the features selection techniques but the SVM with TF-IDF surpasses all the computations and give the best result. SVM is a binary classifier as it creates a hyperplane between data of two classes. Now the problem arises for classification with multiple classes, so our proposed method is more enhanced when we apply Multi-class SVM which perform very well and give accuracy 95.3%.

We discuss in the paper about how we compute the feature vector and comparison between all the classifiers and why Multi-class SVM perform well in our proposed method section and also state how was the previous work in text-classification in previous work section. And also show the scope of this techniques in future work section.

II. References

1- Li-Ping Jing, Hou-Kuan Huang and Hong-Bo Shi, "Improved feature selection approach TFIDF in text mining," Proceedings. International Conference on Machine Learning and Cybernetics, Beijing, China, 2002, pp. 944-946 vol.2.

2- S.Purohit et al., "Effective Tooling for Linked Data Publishing in Scientific Research," 2016 IEEE Tenth International Conference on Semantic Computing (ICSC), Laguna Hills, CA, 2016, pp. 24-31.